2020.06.01-2020.06.07 周报

团队训练

_wzx27

1.指数生成函数

1.1 基本计数问题

常用于解决一类多重集合计数的问题

设 S 是多重集合$\{n_1\cdot a_1,n_2\cdot a_2,\cdots,n_k\cdot a_k\}$，其中 $n_1,n_2,\cdots,n_k$ 是非负整数。设 h_n 是 S 的 n 排列数。那么数列 $h_0,h_1,\cdots,h_n,\cdots$ 的指数生成函数 $g^{(e)}(x)$ 由下式给出： $$g^{(e)}(x)=f_{u_1}(x)f_{u_2}(x)\cdots f_{u_k}(x)$$ 其中，对于 $i=1,2,\cdots,k$，有 $$f_{u_i}(x)=1+x+\frac {x^2}{2!}+\cdots + \frac {x^{n_i}}{n_i!}$$

又因为 $e^x=1+x+\frac {x^2}{2!}+\cdots$，所以这类因子相乘有很好的性质，便于求解最后的生成函数。

POJ3734

用红、蓝、绿、黄四种颜色给 $1\times n$ 的方块染色，红色和绿色方块的数目必须是偶数。

那么蓝黄的因子为： $1+x+\frac {x^2}{2!}+\cdots + \frac {x^{n}}{n!}+\cdots=e^x$ 红绿的因子为： $1+\frac {x^2}{2!} + \frac{x^4}{4!} \cdots=\frac {e^x+e^{-x}}2$ 所以生成函数为： $g^{(e)}(x)=e^{2x}(\frac {e^x+e^{-x}}2)^2=\frac {e^{4x}+2e^{2x}+2}4$ 从而展开后 $n\gt 1$ 时 $\frac{x^n}{n!}$ 的系数为 $\frac {4^n+2^{n+1}}4$

POJ1322

c 种无限集合，任意选取 n个，相同类型的元素出现两个时就会一起消失，问最后剩下 m 个的概率是多少。

范围:$c\le 100,n,m\le 1e6$

可以概率dp，但好像很麻烦？

考虑生成函数的做法，就是有 m 个集合取了奇数次，另外 $c-m$ 个集合取了偶数次。

$g^{(e)}(x)=(\frac {e^x-e^{-x}}2)^m\cdot (\frac{e^x+e^{-x}}2)^{c-m}=2^{-c}\sum_{i=0}^{m}(-1)^{m-i}{m \choose i}e^{ix}e^{-(m-i)x}\cdot \sum_{j=0}^{c-m}{c-m \choose j}e^{jx}e^{-(c-m-j)x}$

然后暴力 $O(c^2logn)$ 就可以每一对 $(i,j$ 对 $\frac {x^n}{n!}$ 的系数的贡献 $(-1)^{m-i}{m \choose i}{c-m \choose j}(2i+2j-c)^n$。

```
#include<iostream>
```

```
#include<algorithm>
#include<iomanip>
#define ll long long
#define pii_ pair<int,int>
#define mp_ make_pair
#define pb push_back
#define fi first
#define se second
#define rep(i,a,b) for(int i=(a);i<=(b);i++)
#define per(i,a,b) for(int i=(a);i>=(b);i--)
#define show1(a) cout<<#a<<" = "<<a<<endl
#define show2(a,b) cout<<#a<<" = "<<a<<"; "<<#b<<" = "<<b<<endl
using namespace std;
const ll INF = 1LL<<60;
const int inf = 1<<30;
const int maxn = 2e5+5;
inline void fastio() {ios::sync_with_stdio(false);cin.tie(0);cout.tie(0);}

double qpow(double a,ll b) {double
s=1;while(b){if(b&1)s=s*a;a=a*a;b>>=1;}return s; }
int c,n,m;
double C[105][105];
int main()
{
    fastio();
    rep(i,0,100) C[i][0]=1;
    rep(i,1,100) rep(j,1,i) C[i][j] = C[i-1][j-1]+C[i-1][j];
    while(cin>>c && c){
        cin>>n>>m;
        if(m>c||m>n||(n-m)&1) {cout<<"0.000"<<endl;continue;}
        double ans = 0.0;
        rep(i,0,c-m){
            rep(j,0,m){
                double t = 1.0*(2*i+2*j-c)/c;
                if((m-j)&1) ans -= qpow(t,n)*C[c-m][i]*C[m][j];
                else ans += qpow(t,n)*C[c-m][i]*C[m][j];
            }
        }
        ans *= 1.0*C[c][m]/qpow(2,c);
        cout<<setprecision(3)<<ans<<endl;
    }
    return 0;
}
```

Infinity37

Zars19

本周推荐